Using PHP

DataDirect Connect® for ODBC

Introduction

PHP is a server-side scripting language that has proven its popularity and success through explosive growth. In the last five years, the number of domains using PHP to generate Web pages has grown from under two million to nearly 20 million (http://www.php.net/usage.php), and there are no signs of that trend slowing down. PHP long ago moved from a hobbyist tool for creating dynamic Web pages to a commercially viable option for large Web sites and enterprise-level applications including Google, Yahoo, and SugarCRM.

Most of the PHP applications in existence today rely on a relational database to store the information that is used to create dynamic Web pages. Connecting to these databases in a manner that can handle a high volume of transactions is crucial to ensure availability and performance of Web-based applications. DataDirect Connect® *for* ODBC can provide the Web developer access to all of the major databases, which gives the maximum amount of flexibility for applications. DataDirect Connect *for* ODBC provides connectivity from UNIX/Linux platforms to Microsoft SQL Server and is also the only suite of ODBC drivers to offer wire protocol solutions that do not require database vendor client libraries such as Oracle Net or DB2 Connect. In addition to eliminating the installation of the database client libraries and reducing the associated maintenance tasks, wire protocol drivers also offer performance advantages by reducing the network communication layers between the application and database.

This article explains how to use DataDirect Connect *for* ODBC with PHP to maximize a PHP deployment on either UNIX/Linux or Windows. Benefits such as improved performance, reduced server resources, and the elimination of database client libraries are why DataDirect Connect products have been deployed by more than 270 commercial software vendors and in thousands of corporate applications worldwide.

Connecting to a Database and Executing SQL with PHP

The PHP function `odbc_connect()` is used to establish a connection to a database. This function accepts three required and one optional argument:

```
resource odbc_connect (string dsn, string user, string password
[, int cursor_type])
```

The *dsn* argument must be a Data Source Name. The *user* and *password* arguments identify the user. Finally, the optional *cursor_type* argument is used to specify, if necessary, which cursor method to use. For more information about this argument and PHP ODBC functions, consult the PHP documentation.

Connecting Through a Data Source

On Windows, the ODBC Administrator, available through the Control Panel, provides a GUI for creating a data source that stores connection option values. On UNIX and Linux, you store data sources in a file called odbc.ini. Instructions for defining a data source can be found in the *DataDirect Connect and Connect XE for ODBC User's Guide and Reference*.

If, for example, you created a data source named DataDirectMSSQL for connecting to a Microsoft SQL Server database, and had a user name of sa and a password admin, you would use the following syntax in PHP:

```
$conn=odbc_connect("DataDirectMSSQL", "sa", "admin");
```

The $conn variable is a resource handle used to refer to this connection in subsequent code.

SQL Examples

The following examples demonstrate making a connection and executing SQL statements. Recall that in the previous examples, the $conn variable stores the information for connecting to a database.

Create Table and Insert Values

This code creates a new EMPLOYEE table and inserts a record:

```
$stmt="CREATE TABLE EMPLOYEE (LAST_NAME varchar(20),
  FIRST_NAME varchar(20))";

$resultset=odbc_exec($conn,$stmt);

$stmt="INSERT INTO EMPLOYEE VALUES('Griffin','Peter')";

$resultset=odbc_exec($conn,$stmt);
```

Query Table and Display Results

This code executes a query and displays the results as an HTML table:

```
$resultset=odbc_exec($conn, "SELECT * FROM EMPLOYEE");

odbc_result_all($resultset,"border=1");
```

Persistent Connection

To maximize performance, PHP supports keeping the database connection open after a script has completed. Subsequent calls with identical connection parameters can use an existing connection, which results in faster execution time and reduced system use. The only change necessary in your code is to call `odbc_pconnect` instead of `odbc_connect`. Note that this has an effect only when PHP is running as a server module and not a cgi process.

Connection Pooling

You are encouraged to use connection pooling for optimal performance. More information is available in the *DataDirect Connect and Connect XE for ODBC User's Guide and Reference*.

PHP on Windows

A common PHP configuration in Windows environments uses IIS 5.0, PHP 5.0, and Microsoft SQL Server 2000. The following table lists sources for acquiring the necessary software:

Install DataDirect Connect *for* ODBC

PHP 5.0	http://www.php.net/downloads.php
IIS 5.0	Included in Windows distributions
Microsoft SQL Server 2000	http://www.microsoft.com/sql
DataDirect Connect *for* ODBC	http://www.datadirect.com/download/index.ssp

Install the DataDirect Connect *for* ODBC drivers. Fifteen-day evaluation copies are available at http://www.datadirect.com/download/index.ssp. An installation guide and user's guide and reference is available online at:

http://www.datadirect.com/techres/odbcproddoc/index.ssp.

Follow the instructions in the *DataDirect Connect and Connect XE for ODBC User's Guide and Reference* for configuring a data source through the ODBC Administrator.

Install IIS 5.0 and Microsoft SQL Server 2000

Install IIS 5.0 and Microsoft SQL Server 2000 on your system. Consult www.microsoft.com for IIS and Microsoft SQL Server 2000 installation instructions. No modifications are necessary to work with PHP and DataDirect Connect *for* ODBC.

Install PHP 5.0

Install PHP 5.0 on your system. When you install PHP 5.0 with IIS, we strongly recommend that you use the ISAPI module method described at http://www.php.net/manual/en/install.windows.iis.php. After PHP 5.0 has been installed and configured, the next step is to verify that ODBC support is included by examining the output of a `phpinfo()` function inside of a PHP script:

```
<?php phpinfo(); ?>
```

If ODBC support is included, the output of the `phpinfo()` function will include an ODBC section with the following settings:

ODBC Support	**Enabled**
Active Persistent Links	0
Active Links	0
ODBC library	Win32

Note that ODBC support is included by default in the binary distributions of PHP.

PHP on UNIX/Linux

A common PHP configuration in UNIX/Linux environments uses PHP 5.0 running on Red Hat Linux 9.0 with Apache 2.0. The following table lists sources for acquiring the necessary software:

PHP 5.0	http://www.php.net/downloads.php
Red Hat 9.0	http://www.redhat.com and other sites such as http://www.linuxiso.org/distro.php?distro=7
Apache 2.0	http://httpd.apache.org/download.cgi
DataDirect Connect *for* ODBC	http://www.datadirect.com/download/index.ssp

Install DataDirect Connect *for* ODBC

Install the DataDirect Connect *for* ODBC drivers. Fifteen-day evaluation copies are available at http://www.datadirect.com/download/index.ssp. An installation guide and user's guide and reference is available online at:

http://www.datadirect.com/techres/odbcproddoc/index.ssp.

Follow the instructions in the *DataDirect Connect and Connect XE for ODBC User's Guide and Reference* for setting environment variables and configuring a data source through the odbc.ini file.

Build PHP 5.0

Instructions for building PHP 5.0 as an Apache 2 shared module can be found at http://www.php.net/manual/en/install.unix.apache2.php. These instructions along with the following steps will guide you through the build process.

1. Follow the instructions up to the statement: "Now, configure your PHP."

2. Set the environment variable ODBC_HOME to point to your installation of the DataDirect Connect *for* ODBC drivers, for example:

    ```
    ODBC_HOME=/opt/DataDirect/ODBC; export ODBC_HOME

    ODBCINST=/opt/odbc32v51/odbcinst.ini;export ODBCINST
    ```

3. In the $ODBC_HOME/include directory, create a file named odbc.h with the following contents:

    ```
    #include <sql.h>
    #include <sqlext.h>
    #include <odbcinst.h>
    ```

4. Set the following environment variables to include DataDirect header (.h) files:

    ```
    CPPFLAGS="-I$ODBC_HOME/include"; export CPPFLAGS
    ```

5. Set the following environment variables to link DataDirect libraries:

    ```
    CUSTOM_ODBC_LIBS="-L$ODBC_HOME/lib -lodbc -lodbcinst";
    export CUSTOM_ODBC_LIBS
    ```

6. Add the following directive to the "configure your PHP" step found in the PHP installation instructions:

    ```
    ./configure --with-apxs2=/home/apache2/bin/apxs

    --prefix=/home/php5 --with-custom-odbc=$ODBC_HOME
    ```

7. Continue with the PHP installation instructions for the `make` command until you reach the `apachectl start` command. Before starting Apache, add the following environment variables to the envvars file found in the bin directory of Apache. For example:

    ```
    ODBCINI="$ODBC_HOME/odbc.ini"

    ODBCINST="$ODBC_HOME/odbcinst.ini"
    ```

    An alternative is to specify the environment variables in each PHP script, such as:

    ```
    <?php
    putenv("ODBCINI=/opt/DataDirect/odbc/odbc.ini");
    putenv("ODBCINST=/opt/DataDirect/odbc/odbcinst.ini ");
    //the rest of the connection code…
    ```

    Consult the Apache and PHP documentation for more information on setting environment variables.

8. An optional step is to verify that the PHP module can load all of the dynamic modules it requires by running an `ldd` command against the libphp5.so file found in the modules directory of your Apache home directory. The output should include references to the libodbc.so, libivicu*xx*.so, and libodbcinst.so shared object files that are found in the $ODBC_HOME/lib directory.

9. After starting your Apache server, you can verify your PHP installation by running a script with the `phpinfo()` function such as:

```
<?php phpinfo(); ?>
```

10. If ODBC support is included, the output of the `phpinfo()` function will include an ODBC section with the following settings:

ODBC Support	**Enabled**
Active Persistent Links	0
Active Links	0
ODBC library	custom
ODBC_INCLUDE	-$ODBC_HOME/include
ODBC_LFLAGS	-$ODBC_HOME/lib
ODBC_LIBS	-L$ODBC_HOME/lib -lodbc –lodbcins

Note: $ODBC_HOME will be replaced with the value you specified at build time.

Summary

PHP is a robust scripting language useful for everything from small projects to enterprise application development. By using DataDirect Connect *for* ODBC drivers, you can easily write portable PHP code for all major operating systems and platforms. DataDirect Connect *for* ODBC offers both performance and scalability benefits through the use of its unique wire protocol architecture. Wire protocol architecture not only improves performance and scalability, but removes the need for vendor client libraries, which results in easier deployment and maintenance of your applications.

DataDirect Connect *for* ODBC drivers are the most widely used third-party ODBC drivers in the software market today and are backed by an award-winning technical support team. For more information, please see http://www.datadirect.com/products/odbc/index.ssp.

We welcome your feedback! Please send any comments concerning documentation, including suggestions for other topics that you would like to see, to:

docgroup@datadirect.com

Last revised: April 2008

FOR MORE INFORMATION

800-876-3101

Worldwide Sales

Belgium (French) 0800 12 045
Belgium (Dutch)................0800 12 046
France.............................0800 911 454
Germany 0800 181 78 76
Japan 0120.20.9613
Netherlands..................0800 022 0524
United Kingdom 0800 169 19 07
United States..................800 876 3101

DataDirect Technologies is focused on standards-based data connectivity, enabling software developers to quickly develop and deploy business applications across all major databases and platforms. DataDirect Technologies offers the most comprehensive, proven line of data connectivity components available anywhere. Developers worldwide at more than 250 leading independent software vendors and thousands of corporate IT departments rely on DataDirect® products to connect their applications to an unparalleled range of data sources using standards-based interfaces such as ODBC, JDBC™ and ADO.NET. Developers also depend on DataDirect to radically simplify complex data integration projects using XML products based on the emerging XQuery and XQJ standards. DataDirect Technologies is an operating company of Progress Software Corporation (Nasdaq: PRGS), a US$300+ million global software industry leader. Headquartered in Bedford, Mass., DataDirect Technologies can be reached on the Web at http://www.datadirect.com or by phone at +1-800-876-3101.